

**Amendments to the Specification**

Please replace the Abstract with the replacement abstract included as an appendix.

Applicant makes the following amendments to the identifying numerals of Figure 1 in the corresponding text. No new matter has been added.

Please replace paragraphs [0026] - [0034] with the following amended paragraphs:

[0026] FIG. 1 illustrates an exemplary hardware and software environment that could be used to implement the database system described below. In the exemplary environment, a computer system 100 is comprised of one or more processing units (PUs) 102<sub>1..N</sub>, also known as processors or nodes, which are interconnected by a network 104. Each of the PUs 102<sub>1..N</sub> is coupled to zero or more fixed and/or removable data storage units (DSUs) 106<sub>1..M</sub>, such as disk drives, that store one or more relational databases. Further, each of the PUs 102<sub>1..N</sub> is coupled to zero or more data communications units (DCUs) 108, such as network interfaces, that communicate with one or more remote systems or devices.

[0027] Operators of the computer system 100 typically use a-one of the workstations 110<sub>1..A</sub>, a terminal, a computer, or another input device to interact with the computer system 100. This interaction generally comprises queries that conform to the Structured Query Language (SQL) standard, and invoke functions performed by a Relational DataBase Management System (RDBMS) executed by the system 100.

[0028] In one example, the RDBMS comprises the Teradata® product offered by NCR Corporation, the assignee of the present invention, and includes one or more Parallel Database Extensions (PDEs) 112<sub>1..N</sub>, Parsing Engines (PEs) 114<sub>1..B</sub>, and Access Module Processors (AMPs) 116<sub>1..C</sub>. These components of the RDBMS perform the functions necessary to implement the RDBMS and SQL functions, i.e., definition, compilation, interpretation, optimization, database access control, database retrieval, and database update.

[0029] Generally, the PDEs 112<sub>I..N</sub>112, PEs 114<sub>I..B</sub>114, and AMPs 116<sub>I..C</sub>116—are tangibly embodied in and/or accessible from a device, media, carrier, or signal, such as RAM, ROM, one or more of the DSUs 106, and/or a remote system or device communicating with the computer system 100 via one or more of the DCUs 108. The PDEs 112<sub>I..N</sub>112, PEs 114<sub>I..B</sub>114, and AMPs 116<sub>I..C</sub>116—each comprise logic and/or data which, when executed, invoked, and/or interpreted by the PUs 102<sub>I..N</sub>102—of the computer system 100, cause the necessary steps or elements described below to be performed.

[0030] Those skilled in the art will recognize that the exemplary environment illustrated in FIG. 1 is not intended to limit the present invention. Indeed, those skilled in the art will recognize that other alternative environments may be used without departing from the scope of the present invention. In addition, it should be understood that the present invention may also apply to components other than those disclosed herein.

[0031] In an example system, work is divided among the PUs 102<sub>I..N</sub>102—in the system 100 by spreading the storage of a partitioned relational database 118 managed by the RDBMS across multiple AMPs 116 and the DSUs 106<sub>I..M</sub>106—(which are managed by the AMPs 116<sub>I..C</sub>116). Thus, one of the a DSUs 106<sub>I..M</sub> may store only a subset of rows that comprise a table in the partitioned database 118 and work is managed by the system 100 so that the task of operating on each subset of rows is performed by the AMPs 116<sub>I..C</sub>116—managing the DSUs 106<sub>I..M</sub>106—that store the subset of rows.

[0032] The PDEs 112<sub>I..N</sub>112—provide a high speed, low latency, message-passing layer for use in communicating between the PEs 114<sub>I..B</sub>114 and AMPs 116<sub>I..C</sub>116. Further, each of the PDEs 112<sub>I..N</sub>112—is an application programming interface (API) that allows the RDBMS to operate under either the UNIX MP-RAS or WINDOWS NT operating systems, in that each of the PDEs 112<sub>I..N</sub>112—isolates most of the operating system dependent functions from the RDBMS, and performs many operations such as shared memory management, message passing, and process or thread creation.

[0033] The PEs 114<sub>I..B</sub>114—handle communications, session control, optimization and query plan generation and control, while the AMPs 116<sub>I..C</sub>116—handle actual database 118<sub>I..M</sub> table

manipulation. The PEs 114 fully parallelize all functions among the AMPs 116<sub>1..C</sub>116. Both the PEs 114<sub>1..B</sub>114 and AMPs 116<sub>1..C</sub>116 are known as “virtual processors” or “vprocs”.

[0034] The vproc concept is accomplished by executing multiple threads or processes in a PU 102, wherein each thread or process is encapsulated within a vproc. The vproc concept adds a level of abstraction between the multi-threading of a work unit and the physical layout of the parallel processing computer system 100. Moreover, when one of thea PUs 102<sub>1..N</sub> itself is comprised of a plurality of processors or nodes, the vproc concept provides for intra-node as well as the inter-node parallelism.

Please replace paragraphs [0036] - [0037] with the following amended paragraphs:

[0036] The system 100 does face the issue of how to divide a query or other unit of work into smaller sub-units, each of which can be assigned to one of thean AMPs 116<sub>1..C</sub>116. In one example, data partitioning and repartitioning may be performed, in order to enhance parallel processing across multiple AMPs 116<sub>1..C</sub>116. For example, the database 118 may be hash partitioned, range partitioned, or not partitioned at all (i.e., locally processed).

[0037] Hash partitioning is a partitioning scheme in which a predefined hash function and map is used to assign records to AMPs 116<sub>1..C</sub>116, wherein the hashing function generates a hash “bucket” number and the hash bucket numbers are mapped to AMPs 116<sub>1..C</sub>116. Range partitioning is a partitioning scheme in which each of the AMPs 116<sub>1..C</sub>116-manages the records falling within a range of values, wherein the entire data set is divided into as many ranges as there are AMPs 116<sub>1..C</sub>116. No partitioning means that a single one of the AMP 116<sub>1..C</sub>116 manages all of the records.

Please replace paragraphs [0039] - [0044] with the following amended paragraphs:

[0039] Block 200 represents SQL statements being accepted by one of the PEs 114<sub>1..B</sub>114.

[0040] Block 202 represents the SQL statements being transformed by a Compiler or Interpreter subsystem of one of the PEs 114<sub>1..B</sub> into an execution plan. Moreover, an Optimizer subsystem of one of the PEs 114<sub>1..B</sub> may transform or optimize the execution plan in a manner described in more detail later in this specification.

[0041] Block 204 represents one of the PEs 114<sub>1..B</sub> generating one or more “step messages” from the execution plan, wherein each step message is assigned to one of the an AMPs 116 that manages the desired records. As mentioned above, the rows of the tables in the database 118 may be partitioned or otherwise distributed among multiple AMPs 116<sub>1..C</sub>, so that multiple AMPs 116<sub>1..C</sub> can work at the same time on the data of a given table. If a request is for data in a single row, one of the PEs 114<sub>1..B</sub> transmits the steps to one of the AMPs 116<sub>1..C</sub> in which the data resides. If the request is for multiple rows, then the steps are forwarded to all participating AMPs 116<sub>1..C</sub>. Since the tables in the database 118<sub>1..M</sub> may be partitioned or distributed across the DSUs 106<sub>1..M</sub> of the AMPs 116<sub>1..C</sub>, the workload of performing the SQL query can be balanced among AMPs 116<sub>1..C</sub> and DSUs 106<sub>1..M</sub>.

[0042] Block 204 also represents one of the PEs 114<sub>1..B</sub> sending the step messages to their assigned AMPs 116<sub>1..C</sub>.

[0043] Block 206 represents one of the AMPs 116<sub>1..C</sub> performing the required data manipulation associated with the step messages received from one of the PEs 114<sub>1..B</sub>, and then transmitting appropriate responses back to one of the PEs 114<sub>1..B</sub>.

[0044] Block 208 represents one of the PEs 114<sub>1..B</sub> merging the responses that come from the AMPs 116<sub>1..C</sub>.

Please replace paragraph [0048] with the following amended paragraph:

[0048] In this example, the tables 300 and 305 are joined according to equivalence relations indicated in the query. It is the job of the Optimizer subsystem of one of the PEs 114<sub>1..B</sub>, at step 202 of FIG. 2, to select a least costly join plan.